Clock-gating 방법을 사용한 저전력 시스톨릭 어레이 비터비 복호기 구현

류 제 혁[†] · 조 준 동[††]

요 약

본 논문에서는 trace-back systolic array Viterbi algorithm의 저전력 생존 메모리 구현에 관한 새로운 알고리즘을 소개한다. 이 알고리즘의 핵심 아이디어는 trace-back 연산의 수를 줄이기 위하여 이미 생성된 trace-back routes를 재사용하는 것이다. 그리고 trace-back unit의 불필요한 switching activity가 발생하는 영역을 gate-clock을 사용하여 전력소모를 줄이는 것이다. Synopsys Power Estimation 툴인 Design Power를 이용하여 전력소모를 측정하였고, 그 결과 [1]의 논문에서 소개된 trace-back unit에 비하여 평균 40% 전력감소가 있었고, 23%의 면적증가를 보였다.

Low-Power Systolic Array Viterbi Decoder Implementation With A Clock-gating Method

Je-Hyuk Ryu[†] · Jun-Dong Cho[††]

ABSTRACT

This paper presents a new algorithm on low power survivor path memory implementation of the trace-back systolic array Viterbi algorithm. A novel idea is to reuse the already-generated trace-back routes to reduce the number of trace-back operations. And the spurious switching activity of the trace-back unit is reduced by making use of a clock gating method. Using the SYNOPSYS power estimation tool, DesignPower, our experimental result shows the average 40% power reduction and 23% area increase against the trace-back unit introduced in [1].

키워드 : 비터비 복호기(Viterbi Decoder), 이동통신(Mobile Communication), 저전력(Low-power), 하드웨어(Systolic Array), 오류정정 코드(Error Correction Code)

1. Introduction

Viterbi algorithm has been widely applied to the decoding and estimation of information in communication and signal processing units[3]. As the demands for battery powered wireless communications increase, the design of a low power Viterbi decoder[2,4] becomes an increasingly critical concern for hand-held devices.

The Viterbi algorithm can be divided into three functional units: the branch metric unit(BMU), the add-compare-select unit(ACSU), and the survivor memory unit (SMU). While the BMU and ACSU perform arithmetic operations such as addition, multiplication, and maximum / minimum selection, the SMU has to trace a path with the help of decision pointers that were generated in the ASCU.

The design of high performance Viterbi decoders has been investigated intensively in the past three decades. Resently, the low power design of Viterbi decoders has been an important issue for mobile application. We review recent works briefly. Akbari, Jahanian implemented an area efficient, low-power and robust ACS unit for Viterbi decoder in both synchronous and asynchronous architecture.[4] Henning and Chakrabarti proposed adaptively approximate Viterbi decoding by varying truncation length and pruning threshold of the T-algorithm while employing trace-back memory management.[5] Ghanipour and Nabavi investigated

※ This work has been supported by SungKyunKwan University research fund, IDEC and IT-SOC.
† 준 회 원 : 성균관대학교 대학원 정보통신공학부
†† 정 회 원 : 성균관대학교 정보통신공학부 교수
논문접수 : 2004년 9월 17일, 심사완료 : 2005년 1월 3일

re-arranging of arithmetic operation to reduce the number and complexity of computational components.[6] Kawokgy adn Salama implemented an asynchrous Viterbi Decoder[7].

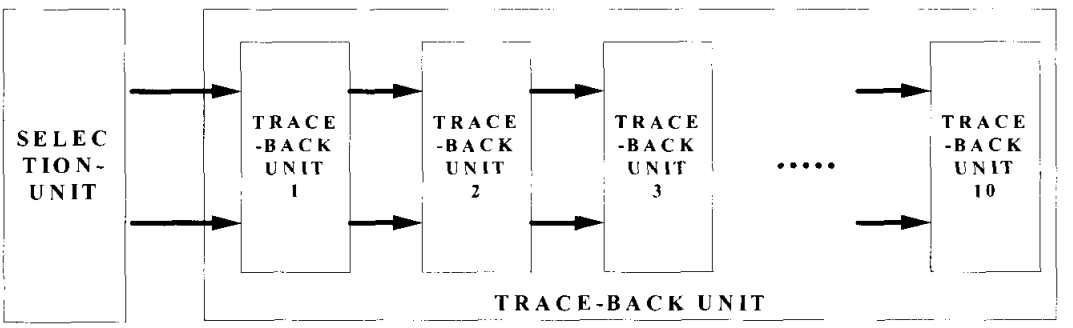
This paper addresses a noble lower-power trace-back method(TBM) for describing the survivor memory management problem in Viterbi decoder that is realized by a trace-back systolic array. A novel idea is to reuse the already-generated trace-back routes to reduce the number of trace-back operations. And the spurious switching activity of the trace-back unit is reduced by using gate clock for reduction of power consumption. To the best of our knowledge, our approach is new and innovative in lower power Viterbi decoder applications.

The following sections in this paper are organized as follows. We first briefly review the conventional trace-back systolic array in Section 2. Section 3 is devoted to the proposed method that was employed to implement a new low power trace back unit. Experimental result and conclusion of this work are in Section 4 and Section 5.

2. Conventional Trace-back Systolic Array Viterbi Decoder

The trace-back movement in systolic array Viterbi decoder carries out decoding in the continuous register array, using the way of pipeline, without delay[5]. As a result, the systolic system can save the decoding time against other decoding methods. (Fig. 1) shows the structure of a systolic Viterbi decoder.

From this system, the trace-back systolic array consists of a Selection Unit(SU) and a Trace-Back Unit(TBU): The SU is composed of a Branch Metric Process Unit(BMPU) in each state and a Min-Selection Unit(MSU) to choose the state with the smallest metric. The TBU is composed of a series of trace-back units. The BMPU computes branch metric by Hamming distance of the basic value of trellis diagram, given by decoder and encoder. Then the branch metric is transferred to the MSU, which determines the state of the smallest metric. After then, decision vector(i.e., a set of the smallest metrics) is transmitted to the TBU. Finally, the TBU traces back to find the decoded information bit.

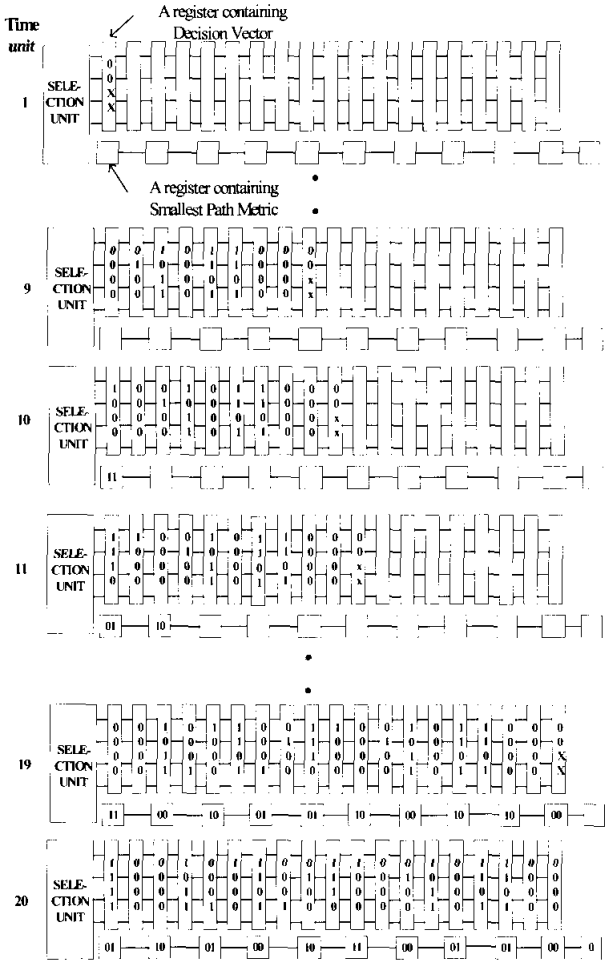


(Fig. 1) The Structure of Systolic Viterbi Decoder

The trace-back algorithm is illustrated in (Fig. 2) where each column shows the decision vector stored in the registers of the trace-back unit. At the time unit 1, the SU computes a decision vector, and then the decision vector is stored in the first register of TBU. At the time unit 2, the SU computes a decision vector, and then the data in the first register is shifted into the second register and the decision vector is stored in the first register of TBU.

This process is repeated until the time unit 9. At the time unit 10, the SU computes a decision vector and selects the state vector with the smallest path metric which is assumed to be (1,1) in our example. Then both decision vector and the smallest path metric are transferred to the register of the register array. Again at the time unit 11, the SU computes a decision vector and selects the smallest path metric, say, (0,1). Then both the decision vector and the smallest path metric are transferred to the register of the register array. At the same time, we compute the trace-back state at the time unit 9 from the smallest path metric (1,1), which returns (1,0). This process is repeated until the time unit 19. At the time unit 20, we obtain the first decoded information bit (0). Note that the decoded information bits are obtained one-by-one sequentially and each trace back unit stores $5K$ bits of data and K bits in the register array. The method of tracking, updating, and storing of the information sequence can be

accomplished simultaneously during a single clock as in [1].



(Fig. 2) The Systolic Array Viterbi Decoder for a(2,1/2) Convolutional Code

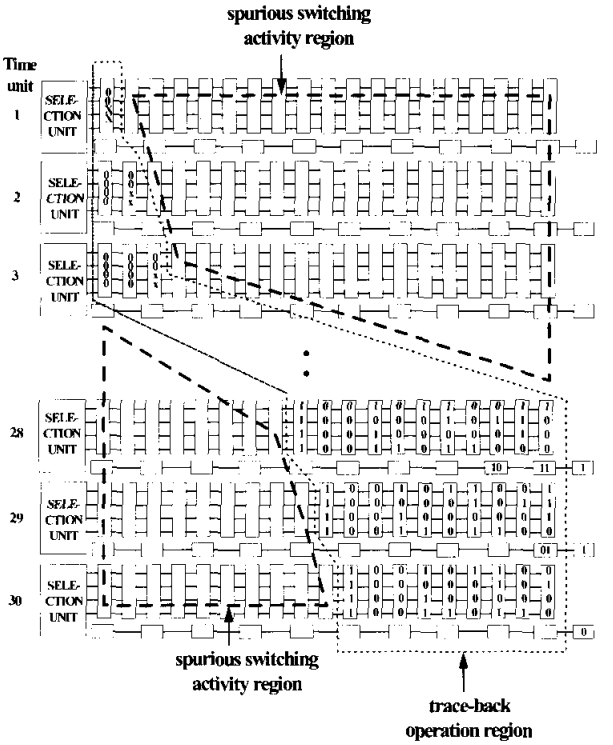
3. LOW POWER TRACE-BACK UNIT

In the following two subsections, we present the clock-gating method in spurious switching activity region and the method of reusing the already-found trace-back routes along with the clock-gating.

3.1. The Clock-Gating Method in Spurious Switching Activity Region

In synchronous circuits, it is possible to stop the clock selectively in portions of the circuit where active computation is not being performed. The presence of a gated clock has a two-fold advantage.

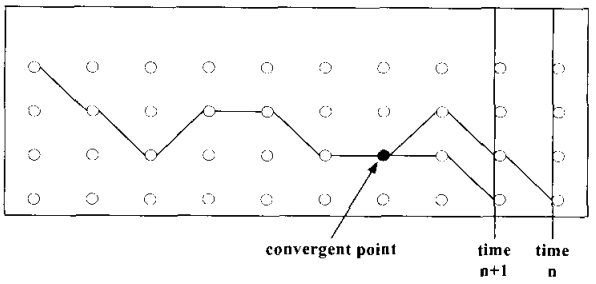
First, when the clock is stopped, no power is consumed in the FSM combinational logic, because its inputs remain constant. Second, no power is consumed in the sequential elements and the gated clock line. As shown in (Fig. 3), the part shown in dotted line(spurious switching activity region) is not an operative region. So we use the gated clock in spurious switching activity region of the trace-back unit to reduce power consumption.



(Fig. 3) Clock-gating of spurious switching activity region in trace-back unit

3.2. Reusing the Already-Found Trace-Back Routes along with Clock-Gating Method

Trace-back unit is the hottest(i.e., power-hungry) spot in a Viterbi decoder due to the frequent memory access charge/discharge activities. Therefore, major portion of power consumption can be reduced if we can revisit the trellis efficiently. Notice that redundant block processing is not attractive if power is the sole consideration. This algorithm is based on the observation that most parts of the two successive trace routes are identical. That is, they converges within a very short distance (refer to (Fig. 4)). Therefore, trace back process is done if we hit the same point of the already-traced routes. And the trace-back control block which has saved already-traced routes creates decoded data by using saved data without doing any trace-back operation any more.

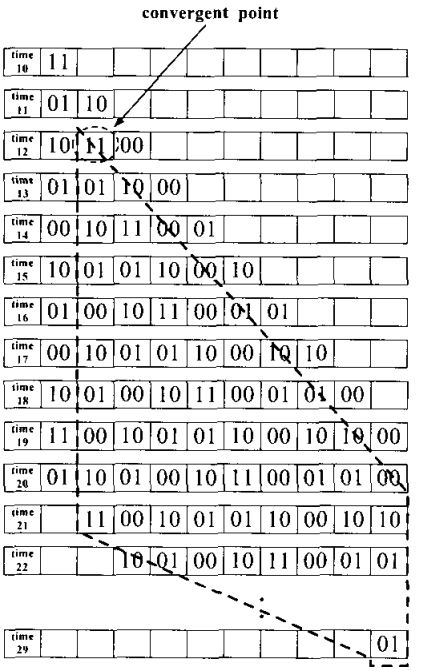


(Fig. 4) Proposed Systolic Array Trace-Back Algorithm based on Trace-Back Routes Reuse

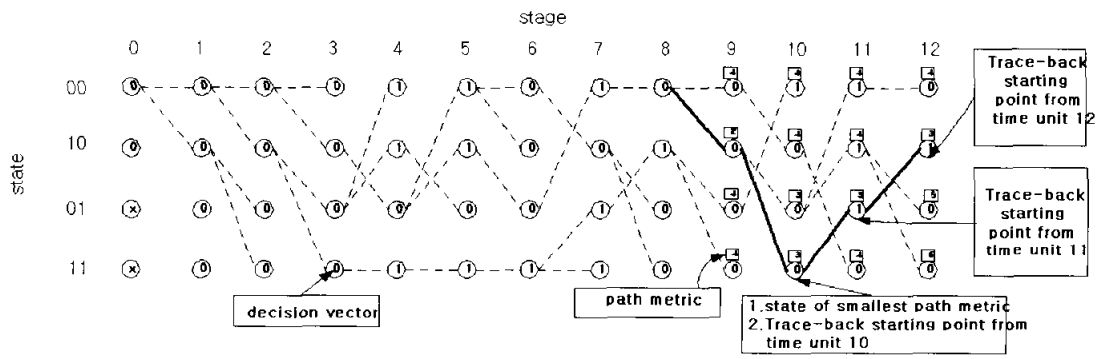
The summary of the algorithm is stated as follows :

1. Initialization: trace back the full 5 times of constraint length trellis and store the path.
2. Loop
   (2-a) Trace and compare : pick up a random initial state and start tracing back. Simultaneously trace the route and compare the each stop with the history of previous route.
   (2-b) Stop and dump the history of previous route: if the result of comparison is the same. Otherwise, repeat step (2-a).
3. Repeat Step (2) for each input signal.

By doing this, we can have the minimum amount of memory access per signal search.



(Fig. 5) Clock-Gating Method in Trace-Back Operation Region with Respect to Convergent Point



(Fig. 6) The sequence of states of the trace-back method

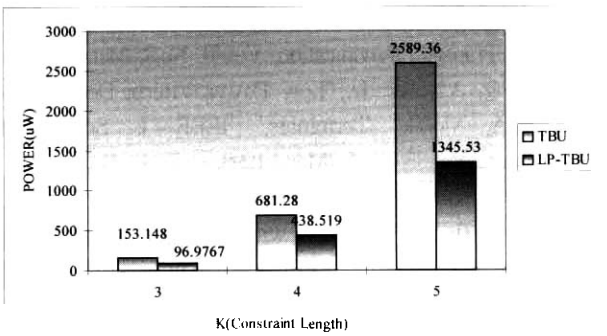
The low power trace-back algorithm based on the reuse of trace-back routes reuse is illustrated in Fig.5 where each entry shows the smallest path metric stored in the registers of the trace-back unit(TBU). The low power trace-back algorithm is summarized in the following(Refer to (Fig. 6)). **At the time unit 10** (corresponding to the stage 10 at (Fig. 6)), SU computes a decision vector and the smallest path metric, the decision vector and the smallest path metric(1,1) are transferred to the first register array of TBU. Also, the smallest path metric(1,1) is stored in the extra register placed at stage 10 to reuse the path in the future. **At the time unit 11**, SU computes a decision vector and the smallest path metric, the decision vector and the smallest path metric(0,1) are transferred to the first register array of TBU. Also, the smallest path metric(0,1) is stored in the extra register. At the same time the trace-back starting from the time unit 10 is performed from the stage 10(1,1) to the stage 9(1,0), and the path metric of the stage 9(1,0) is stored in the extra register. **At the time unit 12**, SU computes a decision vector and the smallest path metric,

the decision vector and the smallest path metric(1,0) are transferred to the first register array of TBU. Also, the smallest path metric(1,0) is stored in the extra register. At the same time the trace-back starting from the time unit 10 traces back from the stage 9(1,0) to the stage 8(0,0) and the path metric of the stage 8(0,0) is stored in the extra register for the path reuse possibly in the near future. At the same time, the trace-back starting from the time unit 11 is performed from the stage 11(0,1) to the stage 10(1,1). **At the time unit 13**, SU computes a decision vector and the smallest path metric, the decision vector and the smallest path metric(0,1) are transferred to the first register array of TBU. Also, the smallest path metric(0,1) is stored in the extra register for the path reuse. At the same time, the trace-back starting from the time unit 10 is performed from the stage 8(0,0) to the stage 7(0,0). Also, the path metric of the stage 7(0,0) is stored in the extra register for the future reuse. At the time unit 12, the stage 12(1,0) is traced back to the stage 11(0,1). When the trace-back starting from the time unit 11 results in the trace-back from the stage 10(1,1) to the stage 9(1,0) simultaneously, the path metric of stage 10(1,1) is compared with the path metric of stage 10(1,1) stored in the extra register for the future path reuse. If the path metric in the stage 10 is same as the path metric stored in the extra register, the path metric of the stage 9 stored in the extra register is copied to the path metric of the stage 9 without running SU. If not the case, with the path metric of the stage 10, the value of the extra register is updated. and the trace-back is performed from the stage 10(1,1) to the stage 9(1,0) again.

After we encounter a convergent point, as shown in (Fig. 5), we do not need to trace and compare the paths within the dotted line. Therefore, we can further reduce power consumption in trace-back operation region by using clock-gating method within the increased spurious switching activity region.

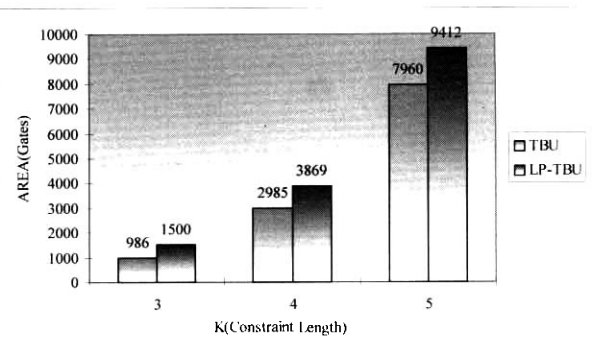
4. Experimental Result

In this paper, we propose a new low power algorithm on the trace back unit of systolic array Viterbi decoder[1]. Reusing the already-generated trace-back routes reduces the number of trace back operations. Also the spurious switching activity is reduced by make use of gated clock to remove unnecessary operations. Our result showed on the average 40% power reduction, but 23% increase in area against the trace-back unit in [1]. Our proposed Viterbi decoder is coded in VHDL. In order to measure the effect of the applied techniques on power consumption, the mentioned techniques is implemented in a circuit. The circuit is simulated with a set of input vector. Transitions in different nodes are recoded in an output file which is read by Synopsys Power estimation tool. The constraints used for our experiment are set as follows: Operating clock is 50Mhz, Target library is LSI10K. As shown in (Fig 7) and (Fig 8), we created a test set by changing the value of K (namely, K=3,4,5) where K denotes the constraint length. When K was 3(resp. 4 and 5), the result showed approximately 36%(resp. 35% and 48%) saving in power and 34%(resp. 22% and 15%) increase in hardware size. (Fig. 6) and (Fig. 7) show the plots of power and area comparison of the conventional trace-back unit with lower power trace-back unit, respectively.



TBU: Trace-back unit, LP-TBU: Lower power Trace-back unit

(Fig. 6) Power comparison



TBU: Trace-back unit, LP-TBU: Lower power Trace-back unit

(Fig. 7) Area comparison

5. Conclusion

We have presented in this paper the modification of the Systolic array trace-back architecture proposed in [1]. Significant power reduction is achieved by Clock-gating Method in Spurious Switching Activity Region and Reusing the Already-Found Trace-Back Routes along with Clock-gating Method while employing trace-back memory management. As comparing with the conventional architectures, 23% area increment and 40% power consumption reduction were obtained. The area overhead can be overcome by using deep sub-micron technologies, thus it is less critical than timing and power consumption. It is difficult to make a head-to-head comparison of power efficiency between the proposed method and other existing methods due to different environments and constraints imposed. Nonetheless some of our techniques can be applied to the other low power designs to reduce power consumption.

Reference

[1] T. K. Truong, A. M. T. Shih, I. S. Reed, E. H.Satorius, "A VLSI Design for a Trace-back Viterbi Decoder," IEEE Trans. Communication, Vol.40, No.3, Mar., 1992.

[2] I. Kang, A.N. W. Jr, "Low-Power Viterbi Decoder for CDMA Mobile Terminals," IEEE J. Solid State Circuits, Vol.33, No.3, Mar., 1998.

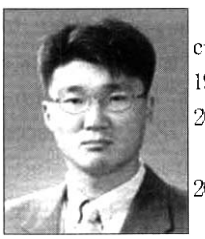
[3] B. W. Kim, J. H. Yang, C. M. Kyung, et al., "MDSP Ⅱ: 16-bit DSP with Mobil Communication Accelerator," in Proc.IEEE Custom Integrated Circuits Conference, pp.2.1.1-2.1.4, 1998.

[4] Akbari, M.K.; Jahanian, A.H.; Naderi, M.; Javadi, B. "Area efficient, low power and robust design for add-compare-select units," Digital System Design, 2004. DSD 2004. Euromicro Symposium on, 31 Aug.-3 Sept. 2004.

[5] Henning, R.; Chakrabarti, C. "Low-power approach for decoding convolutional codes with adaptive Viterbi algorithm approximations," ISLPED '02. Proceedings of the 2002 International Symposium on, pp.12-14 Aug., 2002.

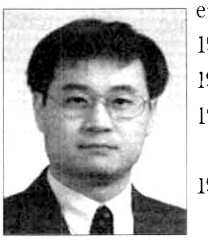
[6] Ghanipour, F.; Nabavi, A.R, "Design of a low-power Viterbi decoder for wireless communications," ICECS 2003. Proceedings of the 2003 10th IEEE International Conference on, Volume: 1, pp.14-17 Dec., 2003.

[7] Kawokgy, M.; Salama, C.A.T., "Low power asynchronous Viterbi decoder for Wireless applications," ISLPED '04. Proceedings of the 2004 International Symposium on, pp.9-11 Aug., 2004.



류 제 혁

e-mail : compro@vada1.skku.ac.kr
1998년 순천대학교 전자공학과(학사)
2000년 성균관대학교 대학원 정보통신공학부(공학석사)
2000년~현재 성균관대학교 대학원 정보통신공학부 재학중(공학박사)
관심분야 : Low Power & High Speed VLSI Design, Communication System, Wireless LAN,



조 준 동

e-mail : jdcho@skku.ac.kr
1980년 성균관대학교 전자공학과(학사)
1989년 Polytechnic University(전산학석사)
1993년 Northwestern University (전기전산학박사)
1995년~현재 성균관대학교 정보통신공학부 교수
관심분야 : VLSI CAD algorithms, lower power designs